Algorithms
Computer Science 140 & Mathematics 168
Instructor: B. Thom
Spring 2004
Homework 13a
Due on 4/29/04 (beginning of class)

1. **[25 points] Network Reliability is NPC!**

   The Network Reliability Problem (NRP) is a famous problem arising in the area of network design. First, a few definitions. We will represent a network as an undirected graph with n vertices $v_1, \ldots, v_n$. We say that two paths in the network are *disjoint* if they have *no vertices* in common except for the endpoints. For example, there can be two disjoint paths from vertex v_{42} to v_{100}, but v_{42} and v_{100} can be the only vertices that these paths have in common (since these vertices are the endpoints of the paths). In this definition, we're assuming that each pair of vertices considered is unique, i.e. $\forall\ v_i, v_j, i \neq j$ (in other words, the number of disjoint paths between a vertex and itself is zero). In the interest of reliability, it is desirable to have multiple disjoint paths between pairs of nodes in the network.

   The Network Reliability Problem (NRP) is defined as follows: Given is an undirected graph with n vertices $v_1, \ldots, v_n$, an $n \times n$ symmetric matrix R of positive integers, and a positive integer b. The question is whether there exists a subset S containing exactly b edges in G such that for *each pair* of vertices v_i and v_j there exist at least r_{ij} *vertex disjoint* paths from v_i to v_j such that all paths are constructed using *only* edges from set S?

   One could imagine NRP being very useful when designing a specific network configuration. For example, if you had n nodes you wanted to connect together, you could begin by assuming you were going to link each node to every other (i.e. build a complete graph, $$$!), and then, given some desired minimal number of disjoint paths between node pairs, ask "How few physical links can be used to still achieve a desired disjointness?"

   Some advice:

   (a) The fact that this problem is in NP is less obvious than any others we've seen so far—when demonstrating this fact, you should provide insight into how a solution can be verified in polynomial time.

   (b) You will find it very useful to first prove (so you can then use) the following lemma: For any graph with E edges, the sum of all its vertices' degrees is exactly $2 \cdot |E|$.

   (c) This problem is difficult. Start early and come speak to me if you need help!

   In this problem, you're to do the following:

   (a) Prove that NRP is NP-complete using a reduction from a problem that we've shown in class to be NP-complete.

(b) This problem contains within it another you've seen in a prior network flow homework—the *s-t connectivity* problem. Recall that there you developed a polynomial time algorithm. So, some NRP instances are solvable in polynomial time. Why does this not necessarily mean that $P = NP$?.